

Applicant: Howard Taitel Serial No.: 09/910,170 Filed: July 20, 2001

Page : 6 of 13

In the Claims:

Please amend the claims as follows:

1. (Original) A method comprising:

identifying portions of a model as being either critical to a real-time execution of the model or non-critical to a real-time execution of the model; and

generating code that is capable of real-time execution based on the critical portions of the model.

- 2. (Original) The method of claim 1 wherein non-critical portions are post-processing units.
- 3. (Amended) The method of claim 2 wherein post-processing units are logical units of the model that have no <u>synchronized</u> data outputs that feed non-post-processing sections of the model.
- 4. (Original) The method of claim 1 wherein generating further comprises establishing an inter-process communication link between the code and the non-critical portions of the model.
- 5. (Original) The method of claim 4 further comprising receiving output from the code via the inter-process communications link.
- 6. (Original) The method of claim 5 further comprising executing the code on a target processor.
- 7. (Original) The method of claim 5 further comprising processing the output in the non-critical portions of the model.
- 8. (Original) A computer program product residing on a computer readable medium having instructions stored thereon which, when executed by a processor, cause the processor to:



Applicant: Howard Taitel

Serial No.: 09/910,170 Filed

: July 20, 2001

Page

: 7 of 13

identify portions of a model as being either critical to a real-time execution of the model or non-critical to a real-time execution of the model; and

Attorney's Docket No.: 04899-058001

generate code that is capable of real-time execution based on the critical portions of the model.

9. (Original) A processor and a memory configured to:

identify portions of a model as being either critical to a real-time execution of the model or non-critical to a real-time execution of the model, and

generate code that is capable of real-time execution based on the critical portions of the model.

10. (Original) A method comprising:

specifying a model, the model including sections, a first subset of the sections designated post-processing unit sections and a second subset of the sections designated as core processing unit sections; and

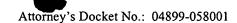
generating software source code for the model with a code generator using the second subset.

- 11. (Original) The method of claim 10 wherein the post-processing unit sections are logical units of the model that have no data outputs that feed core processing unit sections.
 - 12. (Original) The method of claim 10 further comprising:

linking the code to the first subset of sections through an inter-process communication link; and

executing the code on a target processor.

13. (Original) The method of claim 10 wherein specifying the model comprises receiving a user input through a graphical user interface (GUI).



Applicant: Howard Taitel Serial No.: 09/910,170

Filed : July 20, 2001 Page : 8 of 13

14. (Original) The method of claim 10 wherein generating comprises applying a set of software instructions resident in the code generator to the second subset.

- 15. (Original) The method of claim 12 further comprising:
 receiving output from the code via the inter-process communications link; and
 processing the output in the first subset.
- 16. (Original) A system comprising a graphical user interface (GUI) adapted to receive user inputs to specify components of a model, the components containing a first subset of sections designated as post-processing elements of a model and a second subset of sections designated as core elements of the model.
- 17. (Original) The system of claim 16 further comprising an automatic code generator to generate code capable of real-time execution based on the second subset of the sections.
- 18. (Original) The system of claim 17 wherein the second subset includes elements representing essential computational components of the model.
- 19. (Original) The system of claim 16 further comprising a link to provide inter-process communication between the code and the first subset of sections of the model.
- 20. (Original) The system of claim 19 wherein the first subset is non-real time post-processing sections.
- 21. (Original) The system of claim 16 wherein the automatic code generator comprises a set of pre-defined instructions resident in the automatic code generator to generate code corresponding to the second subset.
- 22. (Currently Amended) The system of claim 21 wherein the code is <u>C high level</u> programming language.



Applicant: Howard Taitel Serial No.: 09/910,170 Filed: July 20, 2001

Page : 9 of 13

23. (Original) The system of claim 16 further comprising a compiler for compiling the code for a target processor.

24. (Currently Amended) A method comprising:

receiving user input through a graphical user interface (GUI) specifying a block diagram model, the block diagram model including sections, a first subset of the sections designated post-processing unit sections and a second subset of the section designated as core processing unit sections;

generating software source code for the block diagram model with a code generator using the second subset;

linking connecting the software source code to the first subset via an inter-process communication link; and

compiling the software source code into executable code.

- 25. (Original) The method of claim 24 further comprising executing the executable code on a target processor.
- 26. (Original) A computer program product residing on a computer readable medium having instructions stored thereon which, when executed by the processor, cause the processor to:

specify a model, the model including sections, a first subset of the sections designated post-processing unit sections and a second subset of the section designated as core processing unit sections; and

generate software source code for the model with a code generator using the second subset.

27. (Original) The computer program product of claim 26 wherein the computer readable medium is a random access memory (RAM).





Applicant: Howard Taitel Serial No.: 09/910,170 Filed: July 20, 2001

Page : 10 of 13

28. (Original) The computer program product of claim 26 wherein the computer readable medium is read only memory (ROM).

- 29. (Original) The computer program product of claim 26 wherein the computer readable medium is hard disk drive.
 - 30. (Original) A processor and a memory configured to:

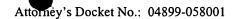
specify a block diagram model, the block diagram model including data having internal pre-defined data storage classes and external custom data storage classes; and generate software source code for the block diagram model with a code generator using the internal predefined data storage classes and the external custom data storage classes.

- 31. (Original) The processor and memory of claim 30 wherein the processor and the memory are incorporated into a personal computer.
- 32. (Original) The processor and memory of claim 30 wherein the processor and the memory are incorporated into a network server residing in the Internet.
- 33. (Original) The processor and memory of claim 30 wherein the processor and the memory are incorporated into a single board computer.
- 34. (Currently Amended) A computer program product residing on a computer readable medium having instructions stored thereon which, when executed by the processor, cause the processor to:

receive user input through a graphical user interface (GUI) specifying a block diagram model, the block diagram model including sections, a first subset of the sections designated post-processing unit sections and a second subset of the section designated as core processing unit sections; and

generate software source code for the block diagram model with a code generator using the second subset;





Applicant: Howard Taite!
Serial No.: 09/910,170
Filed: July 20, 2001
Page: 11 of 13

link-connect the software source code to the first subset via an inter-process communication link; and

compile the software source code into executable code.

35. (Currently Amended) A processor and a memory configured to:

receive user input through a graphical user interface (GUI) specifying a block diagram model, the block diagram model including sections, a first subset of the sections designated post-processing unit sections and a second subset of the section designated as core processing unit sections; and

generate software source code for the block diagram model with a code generator using the second subset;

link connect the software source code to the first subset via an inter-process communication link; and

compile the software source code into executable code.

- 36. (New) The method of claim 5 further comprising executing the code on a host in a target process.
 - 37. (New) The method of claim 7 further comprising displaying the output.
 - 38. (New) The method of claim 7 further comprising archiving the output.
- 39. (New) The method of claim 10 wherein the post-processing unit sections are logical units of the model that have non-synchronized data outputs that feed core processing unit sections.
- 40. (New) The system of claim 18 wherein the second subset is executed in real-time on a target computer.

Attorney's Docket No.: 04899-058001

Applicant: Howard Taitel Serial No.: 09/910,170 Filed: July 20, 2001 Page: 12 of 13

lowy.

41. (New) The system of claim 20 wherein the post-processing sections provide non-synchronized output to the second subset.

42. (New) The method of claim 1 in which post-processing units are logical units of the model that have non-synchronized data outputs that feed core processing unit section.